Resource interoperability revisited

Kerstin Eckart
Universität Stuttgart, IMS
Paffenwaldring 5b
70569 Stuttgart
eckartkn@ims.uni-stuttgart.de

Ulrich Heid
Universität Hildesheim, IWIST
Marienburger Platz 22
31141 Hildesheim
heid@uni-hildesheim.de

Abstract

We present a set of refined categories of interoperability aspects and argue that the representational aspect of interoperability and its content-related aspects should be treated independently. While the implementation of a generic exchange format provides for representational interoperability, content-related interoperability is much harder to achieve. Applying a task-based approach to content-related interoperability reduces complexity and even allows for the combination of very different resources.[1]

1 Introduction

The interoperability of resources is a property which describes how well two resources can interact or be applied together. Thus interoperability is a relevant factor for many processes in the field of natural language processing. Over the years a valuable set of language resources has been created, and this set is still growing. These resources provide the basis for linguistic studies as well as for the development of applications. The more different the resources, the more different are also the forms of interoperability: hence we will start in Section 1.1 with an overview of slightly different notions of interoperability.

Since the creation of a language resource is costly, it is useful to ensure sustainability of the created resources, such that they can be easily reused and extended. One aspect of sustainability is the possibility of existing resources to be in some way combined or utilized together. This way existing and emerging tasks can benefit from the information available in the respective resources. In addition it is important to make transparent where and to which degree resources are interoperable, to increase their acceptance within the user community and to thereby possibly also extend the field in which these language resources are applied.

When we talk about language resources here, we employ a broad definition of the concept. Not only corpora and lexical knowledge bases are subsumed by this notion; also tools for natural language processing, their statistical language models, rule sets, or grammars, as well as data from studies and experiments are to be understood as language resources. Our considerations are intended to capture language data based on different modalities, although the mentioned examples relate to written texts.

1.1 Notions of interoperability

Interoperability of language resources has been discussed in various approaches which focus on different aspects of interoperability and define the concept of interoperability in slightly different ways. In the remainder of this section, we summarize major theoretical viewpoints on the notion of interoperability; below, in Section 4, we will comment on existing implemented applications where interoperability plays a role and discuss them in terms of the theoretical views we will develop in this paper.

Witt et al. (2009) state that the most general notion of interoperability of language resources conveys the idea that these resources are able to interact with each other. Consequently, they classify scenarios of interoperability according to the types of resources to be combined, e.g., applying tools to a corpus vs. combining corpora to create a common subset. Additionally they distinguish between (i) a transfer philosophy of interoperability, where a mapping from the information of one resource to the representation of the other resource is applied, and (ii) an interlingua philosophy of interoperability, where data from both resources are mapped to a new representation that generalizes over both. Ide and Pustejovsky (2010) define interoperability as a measure for the degree to which resources are able to work together and thus aim at an operational definition of interoperability. They describe conditions for interoperability for the following four thematic areas: metadata, data categories, publication of resources and software sharing. Additionally they distinguish between syntactic interoperability and semantic interoperability, adopting these notions from the study of interoperability of software systems and adapting them to the field of computational linguistics. According to them, syntactic interoperability is characterized by properties that ensure that different systems are able to exchange data and to process them either without any conversion or including only a trivial conversion step; while semantic interoperability is the capability to interpret the data in an informed and consistent way. Stede and Huang (2012) focus on linguistic annotation and discuss the role of standard formats for interoperability in an interlingua approach. With respect to the contents of resources, they state that comparability of resources also involves methodology issues, taking the process of creating annotation guidelines into account.

We will adopt the general definition of Witt et al. (2009) that defines interoperability of resources as the ability for these resources to interact, work together or be combined. Our approach also distinguishes between representational and content-related aspects, as Ide and Pustejovsky (2010) do, but we will introduce an additional classification on the content side. Thus our definition of syntactic and semantic interoperability is slightly different from theirs. Like Stede and Huang (2012) we will in particular take the aspect of the combination of linguistic annotations into account.

1.2 Outline

Our contribution is twofold: We will (i) propose refined categories of interoperability aspects (cf. Section 2) and a pertaining classification for interoperability approaches to the combination of different resources (cf. Section 3). Since content-related interoperability, especially with respect to the semantics of the content is most difficult to handle, we cannot expect to be able to solve this issue in a general and comprehensive manner. Therefore we will (ii) introduce an application-oriented proposal for the handling of content-related interoperability issues in a task-based setting and illustrate it with a case study from the task-based combination of syntactic annotations (cf. Section 5). Next to the theoretical set-up in Sections 2 and 3 and the exemplification of our application-oriented proposal in Section 5 we discuss further existing applications in Section 4 in which interoperability plays a role either as the main concern of the approach or as an aspect that has to be dealt with in the actual approach.

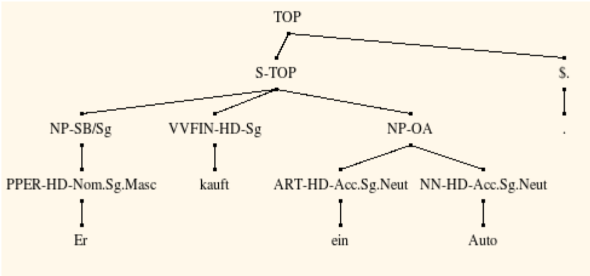
2 Categories of interoperability

To introduce a detailed classification we present an extended and refined concept of interoperability, especially with respect to annotations. On a high level we distinguish between *representational interoperability* and *content-related interoperability* and we subdivide the latter into *syntactic interoperability* and *semantic interoperability*.

Representational interoperability focuses on the different possibilities of representation, i.e. encodings of information. For example, syntactic information is usually structured as a tree, but this tree can be represented by the introduction of brackets to the original input, or it can be encoded in an XML representation, embedded in a figure or arranged in a tabular format. Figure 1 shows three different representations of exactly the same output content produced by the BitPar

(“TOP” (“S-TOP” (“NP-SB/Sg” (“PPER-HD-Nom.Sg.Masc” “Er”))(“VVFIN-HD-Sg” “kauft”) (“NP-OA” (“ART-HD-Acc.Sg.Neut” “ein”)(“NN-HD-Acc.Sg.Neut’ ’ “Auto”)))(“$.” “.”))

(a)



(b)

```
<tokens>
  <token ID="t1">Er</token>
  <token ID="t2">kauft</token>
  <token ID="t3">ein</token>
  <token ID="t4">Auto</token>
  <token ID="t5">.</token>
 </tokens>
<parse>
  <constituent cat="TOP">
    <constituent cat="S–TOP">
      <constituent cat="NP–SB/Sg">
        <constituent cat="PPER–HD–Nom.Sg.Masc"
           tokenIDs="t1"></constituent>
       </constituent>
      <constituent cat="VVFIN–HD–Sg"
           tokenIDs="t2"></constituent>
      <constituent cat="NP–OA">
        <constituent cat="ART–HD–Acc.Sg.Neut"
           tokenIDs="t3"></constituent>
        <constituent cat="NN–HD–Acc.Sg.Neut"
           tokenIDs="t4"></constituent>
       </constituent>
     </constituent>
    <constituent cat="\$."
           tokenIDs="t5"></constituent>
   </constituent>
</parse>
```

(c)

Figure 1: Three representations of the same linguistic content

parser (Schmid, 2004) for sentence (1).

(1) Er kauft ein Auto.
He buys a car.

With respect to linguistic information, i.e., data categories and their structured combination, these three analyses are identical – each of them encodes the same phrase structure tree based on the same grammar and tagsets[2]. Yet, at first sight, it is hard to even see if they are similar. Figure 1(a) is an inline representation of the annotation, where linguistic information is introduced into the original sentence by means of brackets (structure) and tags (part-of-speech, syntactic and morphological information), similar to a well-known representation format of the Penn Treebank (Marcus et al., 1993). Here (``NP-OA'' denotes the start of the noun phrase *ein Auto* which is the direct object of the sentence. Exactly the same linguistic information is represented differently in Figure 1(b). There, we see a graphical representation of the annotated linguistic structure of the sentence. No brackets are applied, but two edges connect the node labelled NP-OA to its children, the parts of the noun phrase. Figure 1(c) is an XML standoff representation of the annotation as an excerpt of the TCF format (Heid et al., 2010). Here the output of the BitPar Parser is represented in its own layer (<parse/>), i.e. separated from the actual tokens (<tokens/>).

While the examples in Figure 1 show how difficult a manual comparison will be, also an automatic comparison of the output would involve either thorough investigation or complex conversion procedures. Thus we claim that representational interoperability is often the first step towards interoperability of resources and that it should not be confused with the linguistically motivated structural decisions reflected in the content. Especially these content-related structural decisions should not get mingled with representational aspects in the process of comparison or conversion.

Content-related interoperability however comprises all linguistically motivated decisions. Here we introduce the additional distinction between

[2]Part-of-speech: ART – determiner, NN – common noun, PPER – personal pronoun, VVFIN – full verb (finite), $. – punctuation symbol at the end of the sentence; syntactic labels: TOP – root, S – sentence, NP – noun phrase, HD – head, OA – direct object, SB – subject; morphological labels: Sg – singular, Acc – accusative, Nom – nominative, Masc – masculine, Neut – neuter

syntactically and semantically motivated differences.

Syntactic interoperability takes structural decisions into account and thus evaluates the similarity of the underlying models: Is the information based on a tree model, i.e. do we have hierarchical categories and no crossing branches, or will we only be able to capture all intended correlations by a directed acyclic graph? Is a node in the tree allowed to have more or less than two children? Are the correlations labelled? To bring out the difference with representational interoperability, in the latter case, the question of where these labels are attached, i.e. to nodes or to edges, would be a representational question. The question important for syntactic interoperability is if correlations are at all intended to include additional information. On a high level, differences with regard to structural interoperability include for example the differences between phrase structure and dependency trees. Figure 2 shows three syntactic annotations for phrase (2)[3]. Figure 2(a) shows a dependency tree based on the output of a parser of the Mate Tools (Bohnet, 2010) and Figures 2(b) and (c) show two phrase structure trees, based on the output of the parser described in Björkelund et al. (2013) (b) and BitPar (c). While in the dependency tree a token is directly connected to its head, phrase structure trees introduce additional nodes for each phrase[4]. Another aspect of syntactic interoperability can be seen in Figure 2(b) and (c). In Figure 2(b) a flat structure is applied, while in Figure 2(c) *deutsche Elf* is considered a phrase of its own.

(2) für die deutsche Elf
for the German eleven
for the German football team

Semantic interoperability focuses on the concepts that are applied within the resources. These are often subsumed by a tagset, where every tag stands for a concept with which parts of the resource can be labelled. A typical example is part-of-speech tagging, where categories such as `noun`, `verb` or `pronoun` are attached to words or word combinations. Distinctions regarding semantic aspects can be found in the annotation guidelines and in the coverage of the single concepts. In the simplest case two different names are applied for the same concept, e.g. `NN` or `N[comm]` for common nouns. More difficulties arise when the same name is applied for different concepts, e.g. when different approaches to dependency syntax use the term `head`, either to refer to a lexical or to a functional head. A further issue is granularity, i.e. cases where a specific concept is applied in one resource, while the it is split into several concepts in another one. The hardest case is one where two concepts only cover part of each other, and no mapping scheme can be applied.

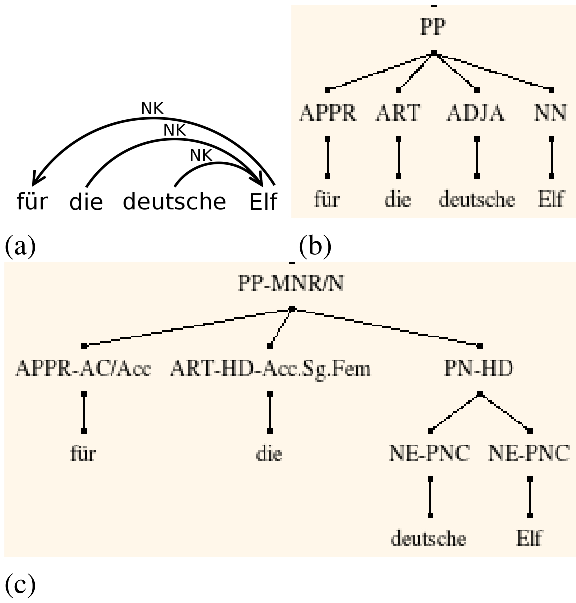


Figure 2: Aspects of syntactic interoperability

Thus when aiming at interoperability of resources, we need to assess the above mentioned three categories individually: representational closeness, syntactic closeness and semantic closeness of the respective resources. Even if these aspects are often interrelated, two resources might show discrepancies to a different degree with respect to each of these categories. Taking this separation into account, it is easier to assess what

[3]The full sentence is: Kevin Kuranyi schoß in Prag beide Tore für die deutsche Elf. *Kevin Kuranyi scored in Prague both goals for the German football team.*

[4]Additional tags: part-of-speech: ADJA – attributive adjective, NE – named entity; syntactic labels: NK – noun kernel, PN – proper noun, PP – prepositional phrase, AC – adpositional case marker, MNR – modifier of a noun phrase to the right, PNC – proper noun component; morphological labels: fem – feminine

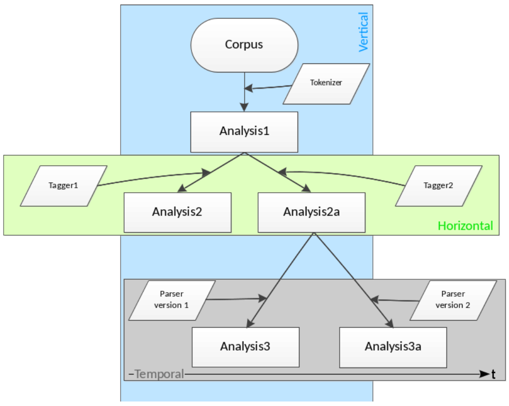


Figure 3: Dimensions of analysis relations as described by Eberle et al. (2012)

is the most beneficial way for a researcher or a project to invest work into achieving resource interoperability.

3 Interoperability: from analysis steps to resources

Since our focus is on interoperability with respect to annotations, we will first discuss which relations can possibly exist between different sets of annotations. Thereafter we introduce classification types for the combination of different resources, which take content-related interoperability into account.

3.1 Interoperable annotations

Linguistic annotations are usually created in an analysis process consisting of several steps. Eberle et al. (2012) describe three dimensions of analysis relations, which are illustrated, for a typical corpus annotation workflow, in Figure 3.

Vertical analysis relations When analysing language data, the analysis steps often reflect a multi-layered structure of language. For textual data the usual (automatic) processing steps include segmentation into sentences and tokens, annotation of part-of-speech tags, generation of syntactic trees representing the structure of each sentence and maybe some further annotation produced e.g. by named entity recognition, coreference resolution, etc. Thus vertical analysis relations exist between 'higher' and 'lower' annotation layers: a 'higher' annotation layer may depend on the information from a 'lower' level. With respect to interoperability, this means that the annotation guidelines and the respective tagset of the 'lower' level need to fit the requirements of the 'higher' level. In an automatic processing chain, these content-related interoperability requirements come with additional ones in terms of representational interoperability: a parser that expects tabular information on segmentation and part-of-speech will not be able to handle plain XML input, even if the tagset is interpretable by the parser.

Horizontal analysis relations For each analysis layer there are several proposals on how to annotate them, starting from different decisions on what a token is, up to the distinction between phrase-structure trees and dependency graphs. Horizontal analysis relations thus exist between alternative analyses from the same linguistic description layer. Since there are many approaches to combine information from different annotations on the same analysis layer (Fiscus, 1997; van Halteren et al., 2001; Björkelund et al., 2013), interoperability is an important factor for the combination of these annotations.

Temporal analysis relations Annotation schemes, annotation tools and knowledge bases may evolve over time. This is catered for by temporal analysis relations. If the same input is annotated by two different instances of the analysis process, the resulting annotation layers might also differ. This relation type is important for the constant development and enhancement of language resources, but is usually rather uncritical with respect to interoperability.

3.2 Interoperable resources

Similar relations exist of course between the resources that are created with vertical, horizontal and temporal analysis steps. An audio corpus, which has been processed by two different systems for the same level of annotation thus yields two resources that are related by a horizontal analysis relation. Documentation on the creation process of a resource therefore often helps to assess if two resources can be applied together. Information about vertical, horizontal and temporal analysis relations can be captured by means of process metadata, stating which input has been processed by which tool(s) in which version.

An important use case for the assessment of interoperability also arises in situations where different resources are to be combined (in a horizontal or a vertical way). In the following we introduce a classification of combination types for such resources, regarding content-related interoperability. Combination type 1 is the case with the highest degree of interoperability, while combination type 3 is a case where neither syntactic nor semantic interoperability are given.

Combination type 1 applies, when two resources are based on the same concepts and the same structural decisions. In this case the resources are fully interoperable with respect to content-related aspects. Examples are different development versions of the same resource or a set of systems taking part in a shared task, where all systems are trained on the same training data.

Combination type 2 applies, when two resources are similar with respect to their structure, but differ with respect to their concepts. Thus semantic interoperability has to be provided while the resources are already syntactically interoperable. Examples are different part-of-speech taggers, that can be applied to the same tokenization; or two lexical resources with a word-based structure but different annotations; or a dependency parser trained on different training sets, that provide for a similar structure with respect to aspects such as projectivity, head-type and coordination.

Combination type 3 applies, when resources differ in structural as well as in semantic criteria. Examples are the combination of prosodic and semantic information based on different segmentations of the primary data; or a query tool for dependency treebanks and a corpus annotated with constituency trees; or a labelled wordnet and a classical lexicon.

4 Existing approaches from new perspectives

In the following some types of realizations from areas like standardization and conversion, shared tasks and evaluation, and processing chains will be classified with respect to our refined concept of interoperability, cf. Section 2, and the classification from Section 3.

Standardization and converter frameworks Stede and Huang (2012) observe that standard formats play an important role in interoperability and tend to be applied as a pivot representation in an interlingua approach, to exchange data between more resource-specific formats without loosing information in the process of mapping. One of these generic exchange formats is GrAF (Ide and Suderman, 2007), the serialization of the Linguistic Annotation Framework LAF (ISO 24612:2012). LAF introduces a layered graph structure, where graphs consist of nodes, edges and annotations. The annotations implement the full power of feature structures and can be applied to nodes and edges alike. All standard annotation layers for linguistic corpora can be mapped onto this model, and since references to the primary data are implemented based on the encoding of their minimal addressable unit, such as characters for a textual representation, or frames for video data, several modalities are covered. LAF/GrAF does thus provide for representational interoperability. Representing each of the three analyses in Figure 1 in GrAF produces an identical result for each of the original representations, and would thus reduce the comparison cost to a minimum. Of course in a typical setting where resources should be combined, the resource annotations are not identical. However mapping them onto a common representation, that is guaranteed to still reflect all resource-specific annotation decisions, helps to bring out the actual content-related differences. To some extent LAF/GrAF can also be used to abstract over features which we relate to syntactic interoperability, such as e.g. condensing annotations from a non-branching path[5] into a combined edge label.

However, by design, LAF itself does not handle semantic interoperability but provides a mechanism for annotation items to link to external concept definitions. Such concept definitions can be set up and referred to in ISOcat[6], a Data Category Registry, based on ISO 12620:2009. There,

[5] Such a non-branching path is e.g. called a unary chain in parsing results.

[6] http://www.isocat.org/

concept definitions are entered in a grass roots approach by the community: if the concept which is needed for a specific resource is not available, it can be entered to the registry. To take care of uncontrolled growth that might result from the grass roots approach, thematic domain groups are supposed to select and recommend specific concepts relevant to thematic domains such as metadata, lexicography, morphosyntax or sign language. Data Category Registries or Concept Registries thus provide most valuable support for semantic interoperability: if two different labels from different resources link to the same concept entry in the registry, they can easily be mapped; if two labels with the same name, but links to different concepts exist in the resources, extra care needs to be taken when the respective resources are to be combined.

In addition, frameworks such as SaltNPepper (Zipser and Romary, 2010) support conversion from one annotation format into another. Salt, the internal meta model of the Pepper converter framework, handles representational differences, and the system also allows to introduce semantic information by external references to ISOcat.

Shared tasks and evaluation projects Shared tasks are usually set up to foster the creation and to enhance and evaluate the quality of language processing systems for a specific task such as machine translation, named entity recognition or dependency parsing. They are however also a platform for the creation of interoperable resources with regard to horizontal relations. In a typical shared task, a certain amount of data is made available that shows the targeted input/output combination. This material can be used to statistically train, or otherwise build a respective system to produce high-value output with respect to the theory or setting the output is based on. At a specific point in time, test data is released, which is processed by the participating systems, and their output is evaluated and ranked by specific metrics. Thus a set of systems emerges, where each system is able to handle the same input data and is aiming to produce the same output information, including the same structure and tagset. These systems are thus possible candidates for an easy combination on the horizontal level.

The project PASSAGE (de la Clergerie et al., 2008), invited parsing systems for French to take part in a collaborative annotation approach of textual data from various sources, including oral transcriptions. The goal was to create a valuable and comprehensive corpus resource for French, by combining the output of different parsing systems in a bootstrapping approach. To be able to combine and merge the annotations, a rather abstract set of categories was defined on which all participating systems could agree. This category set comprised six categories of chunks and fourteen categories of dependencies. On the one hand, this setting brought up an actual use case, where interoperable systems on the same horizontal level were combined to create a new resource. On the other hand, this interoperability was achieved at the cost of abstracting over the content-related differences of the systems, which precisely include the most valuable information in combination approaches.

A similar argumentation applies for the shared tasks regularly conducted in conjunction with the Conference on Natural Language Learning (CoNLL). In 2006 and 2009 the task was on dependency parsing for different languages (Buchholz and Marsi, 2006; Hajič et al., 2009). There the content-related specifications of the system output were not based on the least common denominator like in PASSAGE, but predetermined by the chosen data set for each language. While this allows for more detailed analyses, it still excludes the need for combination of different content-related aspects. However, the CoNLL shared tasks address content-related interoperability in some other respects. Firstly, since the expected output does not only comprise dependency information but also part-of-speech tagging, lemmatization and the identification of morphosyntactic features, the approach thus also fosters interoperability for vertical analysis relations. And secondly, the setup leads to systems that are applicable to many languages. Thereby a language-independent and thus interoperable workflow of training and testing procedures has emerged. Additionally the CoNLL shared tasks gave rise to tabular annotation representations, which have become a de-facto-standard in the field. It thus provides for increasing interoperability on the representational level in horizontal as well as vertical approaches.

An approach to increase content-related, and specifically syntactic interoperability of parser output is embedded in the evaluation methods described by Tsarfaty et al. (2011) and Tsarfaty et al. (2012). In their approach (multi-)function trees are introduced, to which different parse trees can be mapped. In the actual evaluation, tree edit distance is utilized but does not take edits into account which adhere to theory-specific aspects. In multi-function trees, e.g., unary chains over grammatical functions can be condensed into a single edge with a respective label set, thus increasing the syntactic interoperability of the analyses.

Processing chains Processing chains usually implement one path of vertical analysis relations, e.g., starting from the tokenization of primary data and leading up to syntactic and semantic annotations and probably data extraction procedures. Frameworks that implement processing chains are for example UIMA[7] and GATE[8]. A platform for processing chains set up in the context of the CLARIN project[9] is WebLicht[10]. WebLicht lists a set of web services from which the user can build a chain to process some input data. Each web service thereby encodes a natural language processing tool in a so-called wrapper. The output of one web service constitutes the input for another one, until the required annotation level is reached. Thus the processing chain has to deal with three levels of formats: the original input and output format of the underlying tool, the processing format to exchange information between the web services, and, if applicable, an additional output format at the end of the processing chain. In this setting the wrapper ensures content-related interoperability to a certain extent by the way the original tool formats are mapped to the exchange format. Among the different wrappers, representational interoperability is ensured by means of the common processing format that needs to strike a balance between the need for a detailed set of linguistic annotations, and the processing efficiency typically required in a web-based approach.

5 Exemplification: handling interoperability in pieces

In the following we will exemplify how the separation of representational and content-related aspects of interoperability allows to cope with the single aspects individually.

Representational interoperability can effectively be achieved by an interlingua approach based on generic exchange formats such as LAF/-GrAF. Providing for content-related interoperability is even a more difficult task. Different resources are usually based on different approaches and often also on different linguistic theories. Concepts that are important for one resource might not appear at all in another one, or they might partly overlap with concepts utilized in a third resource. Since there is no general ontology to be found that the concepts from all theories and approaches can be mapped to, differences have to be tackled on a case-by-case basis. On the other hand, it is often exactly the heterogeneity of the data that brings upon the benefit of utilizing different resources together. However, for many tasks it is not necessary to provide two fully interoperable resources in order to be able to apply them together.

We illustrate the handling of representational interoperability by means of a relational database approach and the handling of content-related interoperability in a study applying a combination of output from different parsers.

The B3 database (B3DB, Eckart et al. (2010)) is a relational database management system to track workflow aspects and data from computational linguistic projects. The workflow is represented on the macro-layer of the database and the data is structurally represented on its micro-layer. The data structures of the micro-layer are designed on the basis of the LAF/GrAF data model, and are thus generic in the sense that all kinds of different linguistic annotations can be mapped to them, provided these annotations do not exceed the representational power of a graph model. Entering data to the B3DB micro-level thus instantly

[7] http://uima.apache.org/

[8] http://gate.ac.uk/

[9] http://www.clarin.eu/, http://www.clarin-d.de/

[10] Web-based Linguistic Chaining Tool, http://weblicht.sfs.uni-tuebingen.de/weblichtwiki/index.php/Main_Page

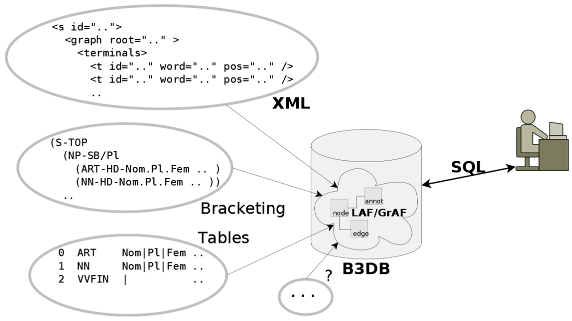


Figure 4: The B3 database as an infrastructure for representational interoperability

provides for representational interoperability of this data.

Figure 4 visualizes the infrastructure setting of the B3DB. Since the data mapping takes place only on the representational level, no explicitly encoded information is lost. A potential user can then conduct SQL-queries on content-related similarities and disagreements of different analyses.

The study by Haselbach et al. (2012) provides an example of task-based handling of interoperability on the level of syntactic analyses. Since Haselbach et al. (2012) are interested in the argument structure of German *nach*-particle verbs, they parsed a web corpus with two different parsing systems: a data-driven state-of-the-art dependency parser trained on news text that provides fully specified analyses (Bohnet, 2010), and a rule-based dependency parser that generates analyses which can be underspecified with respect to head and dependency labels (Schiehlen, 2003). Analysing web data influenced the performance of the systems, but combining information from both systems regarding particle verbs, accusative arguments and dative arguments increased the reliability of the syntactic information and the benefit of the parse results for the overall task. Neither did the actual labels of the different analyses have to be combined, nor was it necessary to resolve all underspecified information. The relevant features for the task were extracted from the output of each system based on its own basic formalism, and only these features were subject to a combination scheme which preferred one or the other analysis, according to the reliability of each parsing system in the respective case.

Such a task-based approach is beneficial in three respects. First it makes it more easy to take the heterogeneity of the information into account and to thus benefit from the differences of the information. Second it supports the handling of information which is specified to a different degree, thus profiting also from underspecified analyses. And third it focuses the effort of the handling of content-related interoperability to the task-related aspects.

6 Conclusion

In this paper, we presented a refined typology of interoperability aspects and argued that the representational aspect of interoperability and its content-related aspects should be treated independently. Regarding content-related aspects, a general and comprehensive solution is not expectable due to the fact that the resources are based on different linguistic theories or approaches. However in many cases such a general or comprehensive solution is not needed to reach a sufficient degree of interoperability for the task at hand. Applying a task-based approach to content-related interoperability reduces complexity to the task-related aspects, and even allows for different combination approaches, depending on the type of task. While it is helpful to have a generic exchange format that provides for representational interoperability in a general fashion, regarding content-related interoperability it might often be more useful to postpone effort of handling it until a specific use case arises.

Acknowledgements

This work was funded by the Deutsche Forschungsgemeinschaft (DFG) in the context of the collaborative research centre SFB 732, project B3.

References

Anders Björkelund, Ozlem Cetinoglu, Richárd Farkas, Thomas Mueller, and Wolfgang Seeker. 2013. (Re)ranking Meets Morphosyntax: State-of-the-art Results from the SPMRL 2013 Shared Task. In *Proceedings of the Fourth Workshop on Statistical Parsing of Morphologically-Rich Languages*, pages 135–145, Seattle, Washington, USA. Association for Computational Linguistics.

Bernd Bohnet. 2010. Top Accuracy and Fast Dependency Parsing is not a Contradiction. In *Proceedings of COLING 2010*, pages 89–97, Beijing.

Sabine Buchholz and Erwin Marsi. 2006. CoNLL-X Shared Task on Multilingual Dependency Parsing. In *Proceedings of CoNLL-X*, pages 149–164, New York.

Eric Villemonte de la Clergerie, Olivier Hamon, Djamel Mostefa, Christelle Ayache, Patrick Paroubek, and Anne Vilnat. 2008. PASSAGE: from French Parser Evaluation to Large Sized Treebank. In Nicoletta Calzolari (Conference Chair), Khalid Choukri, Bente Maegaard, Joseph Mariani, Jan Odijk, Stelios Piperidis, and Daniel Tapias, editors, *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*, Marrakech, Morocco. European Language Resources Association (ELRA).

Kurt Eberle, Kerstin Eckart, Ulrich Heid, and Boris Haselbach. 2012. A Tool/Database Interface for Multi-Level Analyses. In *Proceedings of the eighth conference on International Language Resources and Evaluation (LREC'12)*, Istanbul, Turkey. European Language Resources Association (ELRA).

Kerstin Eckart, Kurt Eberle, and Ulrich Heid. 2010. An Infrastructure for More Reliable Corpus Analysis. In *Proceedings of the Workshop on Web Services and Processing Pipelines in HLT: Tool Evaluation, LR Production and Validation (LREC'10)*, pages 8–14, Valletta, Malta.

Jonathan G. Fiscus. 1997. A Post-Processing System To Yield Reduced Word Error Rates: Recognizer Output Voting Error Reduction (ROVER). In *IEEE Workshop on Automatic Speech Recognition and Understanding*, pages 347–354.

Jan Hajič et al. 2009. The CoNLL-2009 Shared Task: Syntactic and Semantic Dependencies in Multiple Languages. In *Proceedings of CoNLL 2009: Shared Task*, pages 1–18, Boulder, Colorado.

Boris Haselbach, Kerstin Eckart, Wolfgang Seeker, Kurt Eberle, and Ulrich Heid. 2012. Approximating theoretical linguistics classification in real data: the case of German *nach* particle verbs. In *Proceedings of COLING 2012*, Mumbai.

Ulrich Heid, Helmut Schmid, Kerstin Eckart, and Erhard Hinrichs. 2010. A corpus representation format for linguistic web services: the D-SPIN Text Corpus Format and its relationship with ISO standards. In *Proceedings of LREC-2010, Linguistic Resources and Evaluation Conference*, Malta. [CD-ROM].

Nancy Ide and James Pustejovsky. 2010. What Does Interoperability Mean, anyway? Toward an Operational Definition of Interoperability for Language Technology. In *Proceedings of the Second International Conference on Global Interoperability for Language Resources (ICGL 2010)*, Hong Kong, China.

Nancy Ide and Keith Suderman. 2007. GrAF: A Graph-based Format for Linguistic Annotations. In *Proceedings of the Linguistic Annotation Workshop*, pages 1–8, Prague, Czech Republic. Association for Computational Linguistics.

ISO 12620:2009 Terminology and other language and content resources – Specification of data categories and management of a Data Category Registry for language resources.

ISO 24612:2012 Language resource management – Linguistic annotation framework (LAF).

Mitchell P. Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. 1993. Building a Large Annotated Corpus of English: The Penn Treebank. *Comput. Ling.*, 19(2):313 – 330.

Michael Schiehlen. 2003. A Cascaded Finite-State Parser for German. In *Proceedings of EACL 2003*, pages 163–166, Budapest.

Helmut Schmid. 2004. Efficient Parsing of Highly Ambiguous Context-Free Grammars with Bit Vectors. In *Proceedings of the 20th International Conference on Comput ational Linguistics (COLING 2004)*, Geneva, Switzerland.

Manfred Stede and Chu-Ren Huang. 2012. Interoperability and reusability: the science of annotation. *Language Resources and Evaluation*, 46(1):91–94.

Reut Tsarfaty, Joakim Nivre, and Evelina Andersson. 2011. Evaluating Dependency Parsing: Robust and Heuristics-Free Cross-Annotation Evaluation. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 385–396, Edinburgh, Scotland, UK. Association for Computational Linguistics.

Reut Tsarfaty, Joakim Nivre, and Evelina Andersson. 2012. Cross-Framework Evaluation for Statistical Parsing. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 44–54, Avignon, France. Association for Computational Linguistics.

Hans van Halteren et al. 2001. Improving Accuracy in Word Class Tagging through the Combination of Machine Learning Systems. *Computational Linguistics*, 27(2):199 – 229.

Andreas Witt, Ulrich Heid, Felix Sasaki, and Gilles Sérasset. 2009. Multilingual language resources and interoperability. *Language Resources and Evaluation*, 43(1):1–14.

Florian Zipser and Laurent Romary. 2010. A model oriented approach to the mapping of annotation formats using standards. In *Workshop on Language Resource and Language Technology Standards, LREC 2010*, Valetta, Malta.